Parallel Multilevel Fast Multipole Algorithm for GRID computing allowing Full-Wave Electromagnetic Simulations

Jan Fostier

Supervisor(s): Femke Olyslager

Abstract— **This work is concerned with the parallelization of the multilevel fast multipole algorithm (MLFMA) for large scale electromagnetic simulations on distributed memory computers. We focus on techniques suited for low-cost clusters of workstations (COWs) and computational GRIDs. These consist of a number of nodes interconnected through a Gigabit Ethernet network and a fast switch. The algorithm is implemented on a method of moment discretisation of a two dimensional TM electromagnetic scattering problem. Because the parallelization of the MLFMA inherently relies on heavy communication between the nodes special care is taken to avoid congestion of the interconnection network by means of an asynchronous algorithm, i.e. some nodes are communicating while others are performing calculations. This allows us to evaluate large problems such as passive optical devices or electromagnetic compatibility simulations of complex high-frequency electronic equipment with up to several millions of unknowns.**

Keywords—**Electromagnetic scattering, parallel MLFMA**

I. Introduction

ELECTROMAGNETIC simulations have gained considerable interest due to their ability to accurately describe electric and magnetic fields in electrical and optical components. However, such components are typically very large in terms of wavelength, leading to a high number of unknowns N after Method of Moment (MoM) discretisation and thus a high computational cost. Classical methods require $\mathcal{O}(N^2)$ memory and calculation time, making large scale simulations unfeasible. A breakthrough was achieved with the introduction of the Fast Multipole Method (FMM) and its multilevel variant (MLFMA) [1] which reduces this computational complexity to $\mathcal{O}(N \log N)$. However, the problem size that can be solved on a single processor is still relatively small. As cheaper parallel computers become more and more available efforts are made to adapt parallel versions of the MLFMA. Previous work in this field focused on the calculation of the radar cross section of aircraft [2], [3] on dedicated parallel computers, with extremely fast interconnects like Mirinet and Infiniband. In this work, we focus on the development of a generic parallel MLFMA solver for two dimensional electromagnetic scattering problems, aimed at clusters of workstations (COWs) and GRID computers with slower (but cheaper) Gigabit Ethernet interconnects. For these systems the communication time is not negligible compared to the calculation time. An asynchronous algorithm is presented which maximizes overlap between communication and computations. In section II an introduction is given to the MFLMA, followed by an outline of the parallel algorithm in section III.

J. Fostier is with the Department of Information Technology, Ghent University, Sint-Pietersnieuwstraat 41 B-9000 Gent, Belgium. Email: jan.fostier@intec.ugent.be.

II. Method of Moments and the MLFMA

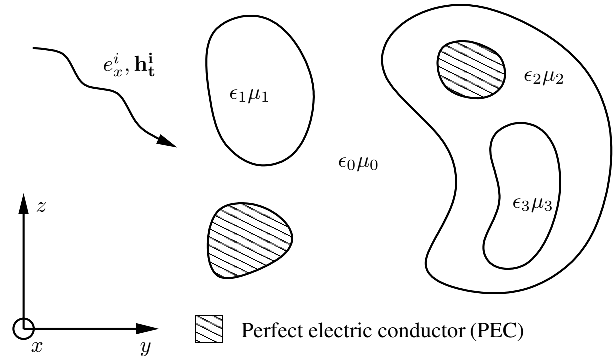

Fig. 1. Objects illuminated by an incoming electromagnetic field.

Consider a number of cylindrical perfect electric conductors (PEC) and dielectric objects of arbitrary shape that are illuminated by an incoming electromagnetic field (see figure 1). These objects can be embedded in other dielectrics. This scattering problem is solved by virtue of a boundary integral equation, so one needs to consider only the electromagnetic fields at the boundary of each object. After discretisation with the MoM technique, this equation leads to a dense set of linear equations of dimension N by N, with N the number of discretisations. For more information regarding this, we refer the reader to [4]. This system can then be solved iteratively requiring an $\mathcal{O}(N^2)$ matrix-vector product in each iterative step. If the number of iterations can be limited this matrix-vector product will be the bottleneck for large N.

The use of the Fast Multipole Method can greatly speed up this matrix-vector multiplication by grouping the unknowns and applying a fast interaction scheme between distant groups giving rise to a $\mathcal{O}(N^{3/2})$ complexity. If done in a multilevel scheme, i.e. by grouping groups into bigger groups and so on, we obtain a tree algorithm (see figure 2). A matrix-vector product then consists out of 3 phases. The first one is the aggregation phase, where for each group the information of the underlying unknowns is represented compactly in a radiation pattern. This information is calculated recursively, i.e. the radiation pattern of a certain parent group is calculated from the radiation pattern of its children. In a second phase the interactions between distant groups at each level are calculated. This is called the translation phase. The final stage, the disaggregation stage, is the converse of the aggregation process, where the translated radiation patterns are recursively sent downwards the tree. The complete process is completely mathematically equivalent with the classical matrix-vector multiplication. The gain in computational complexity lays in the fact that a translation between two groups at a higher level is much more economic than translations between al the respective child groups at lower levels. As mentioned, the MLFMA requires only $\mathcal{O}(N \log N)$ memory and calculation time, making it possible to perform simulations with up to a hundred thousand of unknowns on a single workstation.

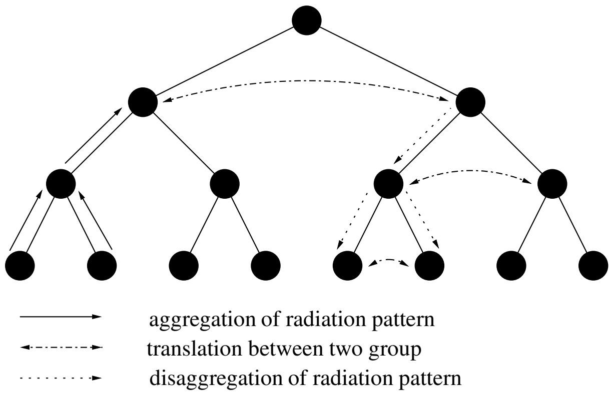

Fig. 2. Schematic representation of the MLFMA. Please note that for clarity only a limited number of arrows is represented.

III. Parallel algorithm

GRID computers and COWs belong to the class of distributed memory computers. Contrary to shared memory computers, where all processors have fast access to a large pool of common memory, distributed memory computers only have access to their own local memory. The processors (nodes) can communicate to each other by sending messages across an interconnection network. These networks are characterized by their latency (delay between sending a byte and its receive) and their bandwidth (the number of bytes that can be sent per second). As mentioned in the introduction we target Gigabit Ethernet networks, which are relatively slow, but also very cheap.

In the parallel algorithm, the tree is distributed across the nodes. This means that each node only has access to a certain number of groups at each level. In each of the three phases, some communication could be required. In the aggregation and disaggregation phases, communication is required if a child belongs to a different node as its parent. During the translation phase, communication is required between groups that reside in different nodes. It is therefore important to maintain as much data locality as possible, i.e. keep groups that require some interaction in one node.

In a synchronous algorithm all processors perform the same operations at the same time on different data. All nodes will start aggregation from one level to another, after which they perform all communications needed for the next aggregation, etc. This separates the computation and the communication phase, which generates bursts of intense network traffic. For Ethernet networks, this approach is not usable, because the collective communication phases generate more traffic than this network can handle, stalling all nodes while waiting for data. This is detrimental for the parallel efficiency.

In our approach, we have developed an asynchronous algorithm, which overcomes this communication bottleneck by overlapping communications and computations. The basic idea is that while some nodes are communicating, other nodes are performing calculations. By spreading communication through time, network congestion is avoided. Furthermore, if a node could not proceed with a certain calculation because data from another node has not yet arrived, it could decide to change the order in which the calculations are performed. This can only happen to some extent because of certain order relations inherent to the MLFMA. Finally, each node will try to prepare and send messages as soon as possible because other nodes are potentially waiting for them.

In figure 3 the parallel efficiency (actual speedup/theoretical speedup) for a simulation of a lens type of object of 10000 wavelenghts in size and requiring 1.9 million of unknowns is shown. It is clearly demonstrated that our asynchronous heuristic outperforms a more naive implementation without communication overlap. These calculations were carried out on a cluster of 12 AMD Opteron 270 processors with a total memory of 24 GByte.

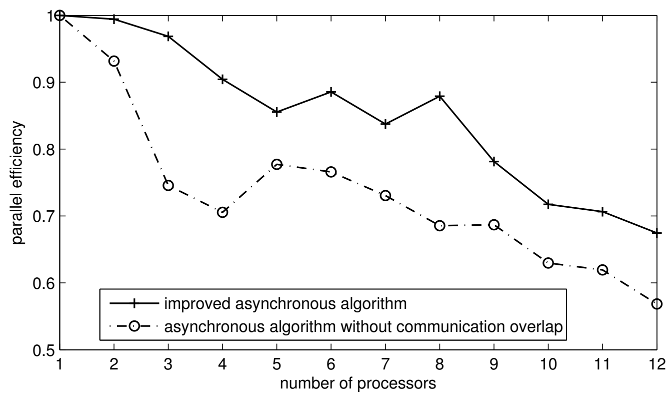

Fig. 3. Efficiency as a function of the number of processors.

IV. Conclusion and Further Research

We have developed an asynchronous parallel MFLMA targeted at GRID computers. This allows us to perform simulations with up to a few million of unknowns. Future research will focus on the extension to three dimensions and a further improvement in parallel efficiency.

V. Aknowledgement

J. Fostier has a grant from the IWT, Belgium.

References

[1] W.C. Chew, J.M. Jin, E. Michielssen, J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*, Boston: artech House, 2001.

[2] S. Velamparambil, W.C. Chew, *Parallelization of Multilevel Fast Multipole Algorithm on Distributed Memory Computers*, Tech. Rep. 13-01, Center for Computational Electromagnetics, University of Illinois at Urbana-Champaign, 2001.

[3] F. Wu, Y. Zhang, Z. Z. Ooo, E. Li, *Parallel multipole fast multipole method for solving large-scale problems* IEEE. Antennas and Propagation Magazine, Vol. 47, No. 4, 2005, pp. 110-118.

[4] J. Fostier, *Elektromagnetische veldberekeningen van extreem grote 2D problemen.*, Master thesis 2005-2006.

Parallel Multilevel Fast Multipole Algorithm for GRID computing allowing Full-Wave Electromagnetic Simulations

Jan Fostier and Femke Olyslager
Supervisor(s): Femke Olyslager

This work is concerned with the parallelisation of the multilevel fast multipole algorithm (MLFMA) for large scale electromagnetic simulations on distributed memory computers. We focus on techniques suited for low-cost clusters of workstations (COWs) and computational GRIDs. These consist of a number of nodes interconnected through a gigabit Ethernet network and a fast switch. The algorithm is implemented on a method of moment discretisation of a two dimensional TM electromagnetic scattering problem.

Because the parallelisation of the MLFMA inherently relies on heavy communication between the nodes special care is taken to avoid congestion of the interconnection network by means of an asynchronous algorithm, e.g. some nodes are communicating while others are performing calculations. This allows us to evaluate large problems such as passive optical devices or electromagnetic compatibility simulations of complex high-frequency electronic equipment with up to several millions of unknowns.

7e FirW Doctoraatssymposium

29 november 2006 :: Het Pand :: Gent :: België